VoiceOver and Making Cocoa Applications Accessible

Session 125

These are confidential sessions—please refrain from filming or taking pictures.

Application Technologies

VoiceOver

Rick Fabrick
Sr. Software Engineer, VoiceOver Team

http://developer.apple.com/wwdc2005/

Session Overview

- VoiceOver
- Demo of VoiceOver
- Importance of accessibility
- Accessorizing your application

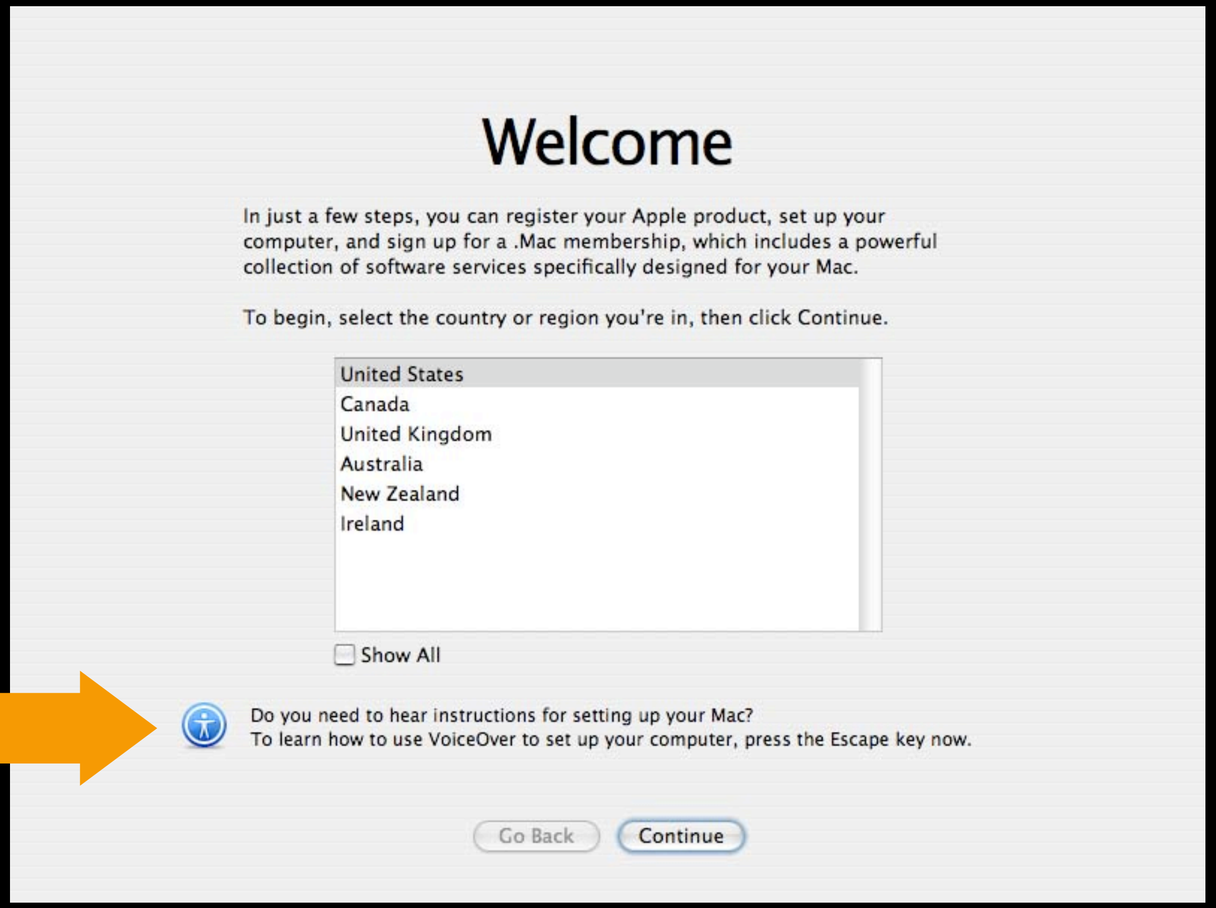
What Is VoiceOver?

- Screen reader software for Mac OS X
- Speaks screen activity and desktop orientation
- Provides keyboard access to everything

Built In, not Bolted On!

Available During Installation

Setup Assistant Tutorial



Available During User Log In

Available Throughout Mac OS X

Many Tiger Apps Already Accessible

Available Everywhere!

Who Will Benefit?

- Users who are blind, visually impaired, or learning disabled
- Anyone who collaborates with the visually impaired
- Aging population

You!

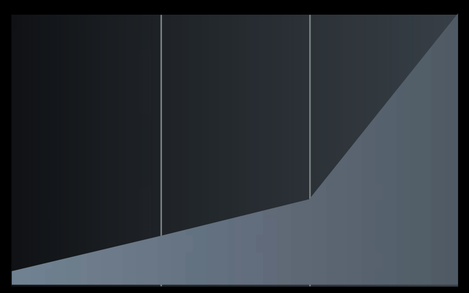
Education

Government

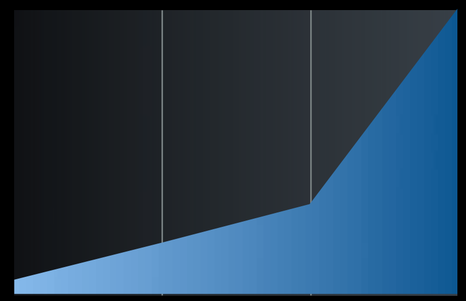
Enterprise

Section 508

Accessible Applications



Your market share



Apple’s market share

VoiceOver Demo

http://developer.apple.com/wwdc2005/

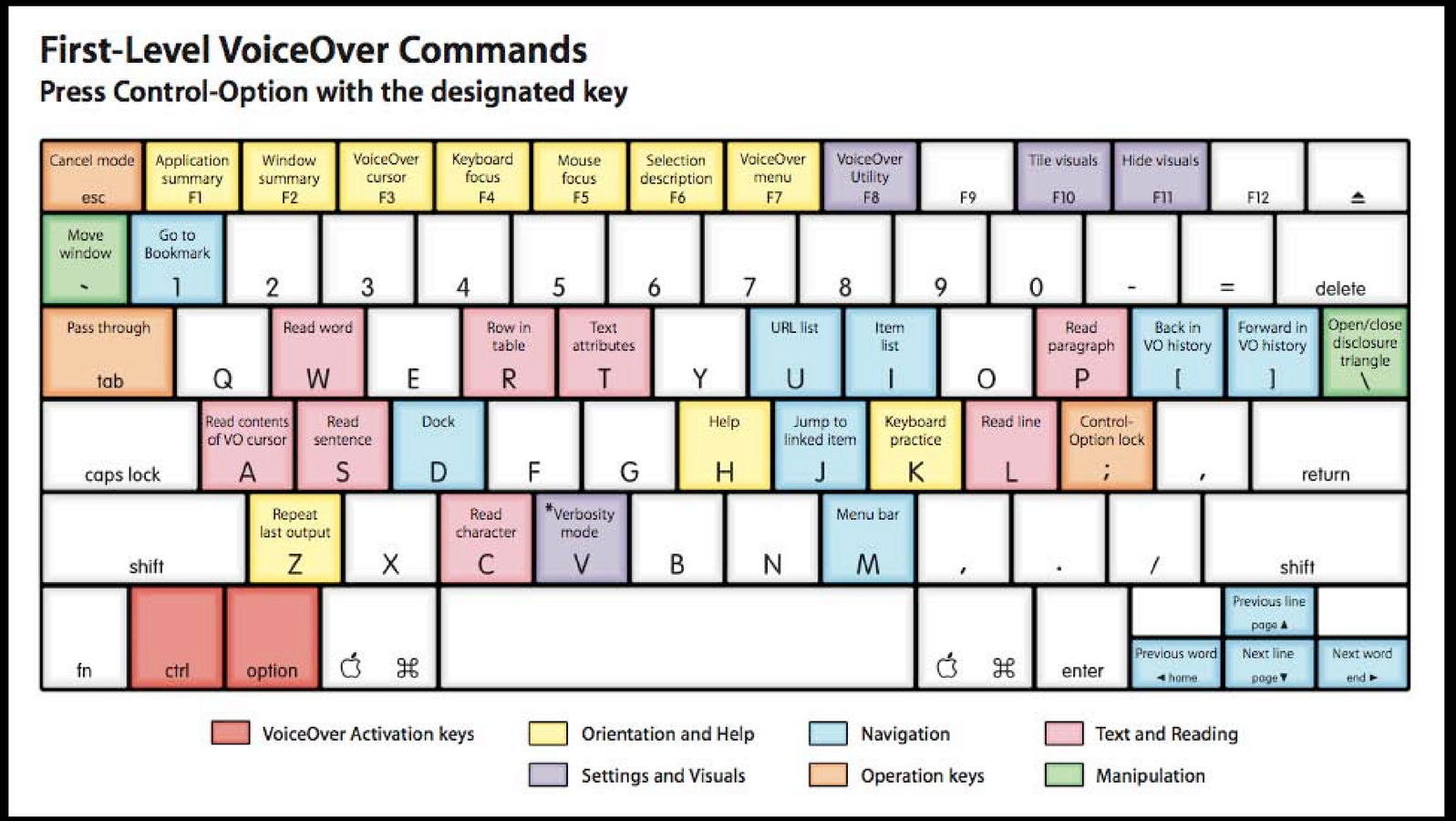
What Should I Do Now?

- Support accessibility APIs
- Hook up full keyboard navigation
- Provide keyboard alternatives to mouse operations
- Tools to help:

Accessibility Verifier

Accessibility Inspector

Test with VoiceOver



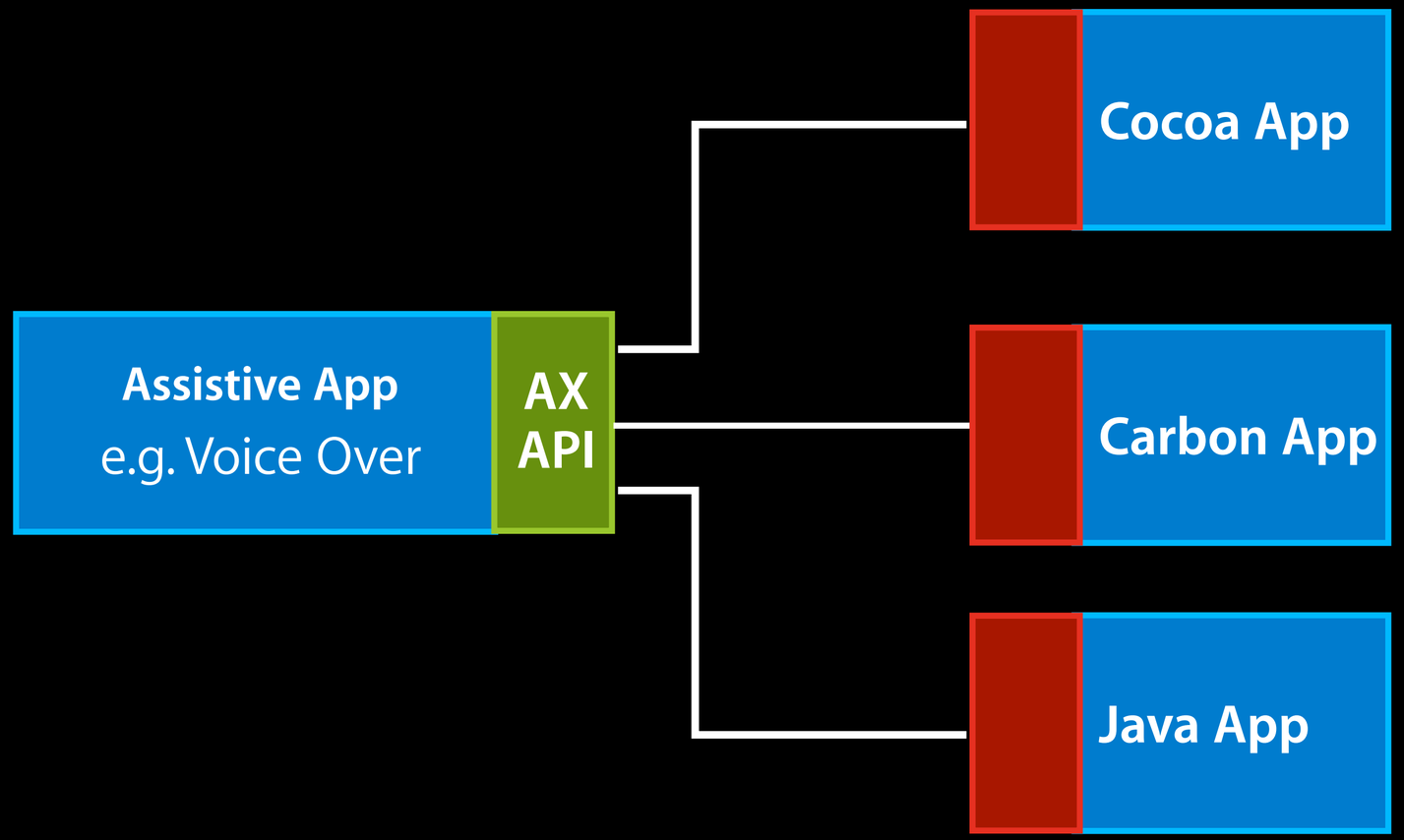
http://www.apple.com/accessibility/voiceover/

Making Cocoa Applications Accessible

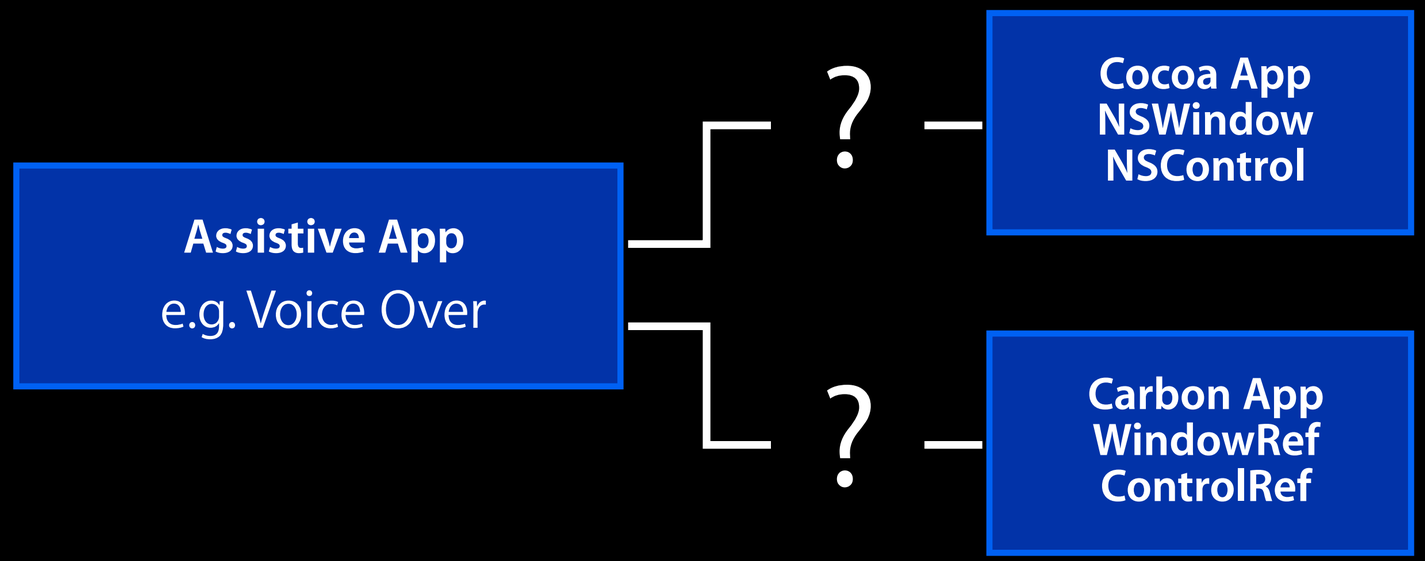
Mike Engber
job title (36 chars max)

http://developer.apple.com/wwdc2005/

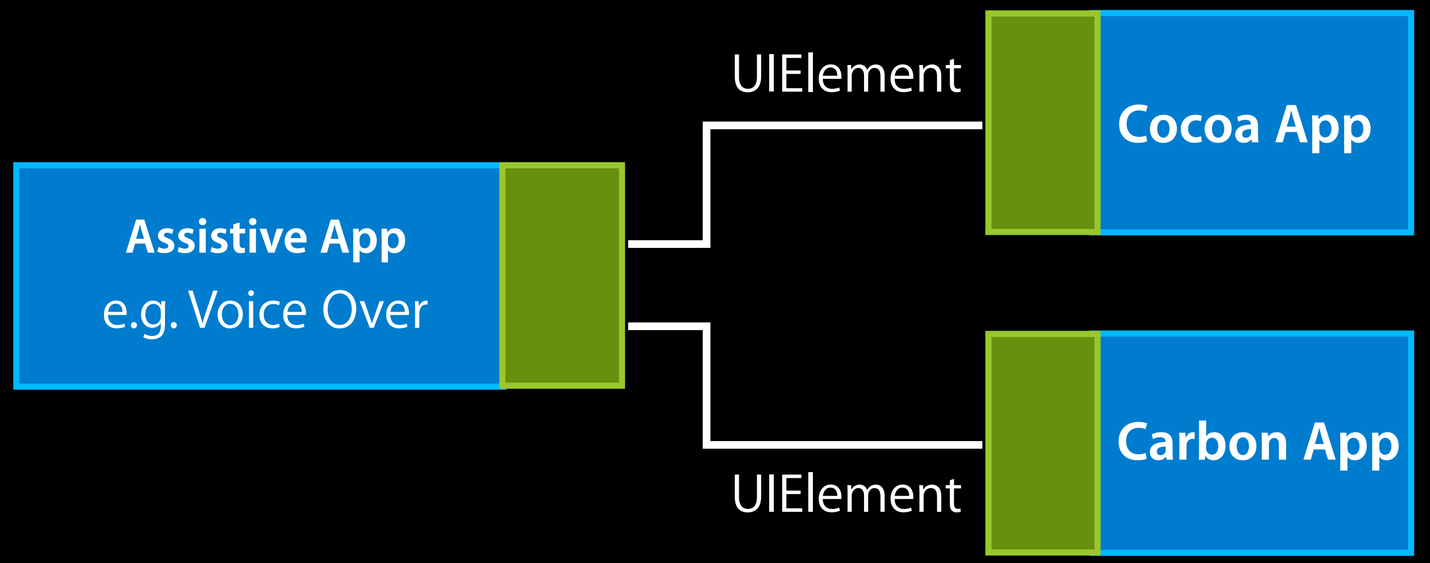
Accessibility Overview



UI Representation
Problem



UI Representation
Solution



UI Elements

- Hierarchy of UIElements
- Attributes
  - kAXTitleAttribute
  - kAXValueAttribute
  - kAXChildrenAttribute
- Actions
  - kAXPressAction
  - kAXIncrementAction

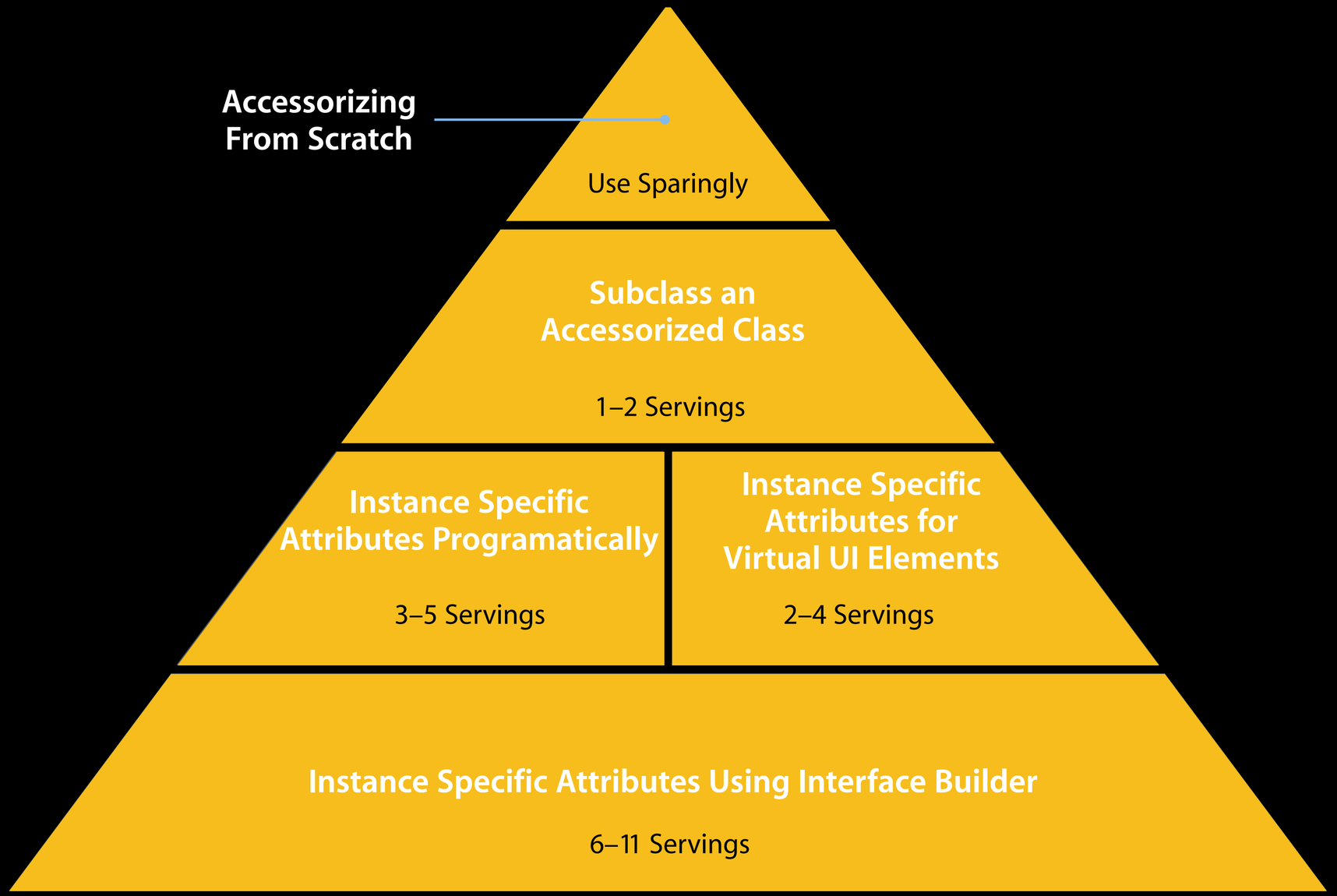
Demo

Accessibility Inspector

http://developer.apple.com/wwdc2005/

Effort to Support Accessibility

Accessibility Pyramid



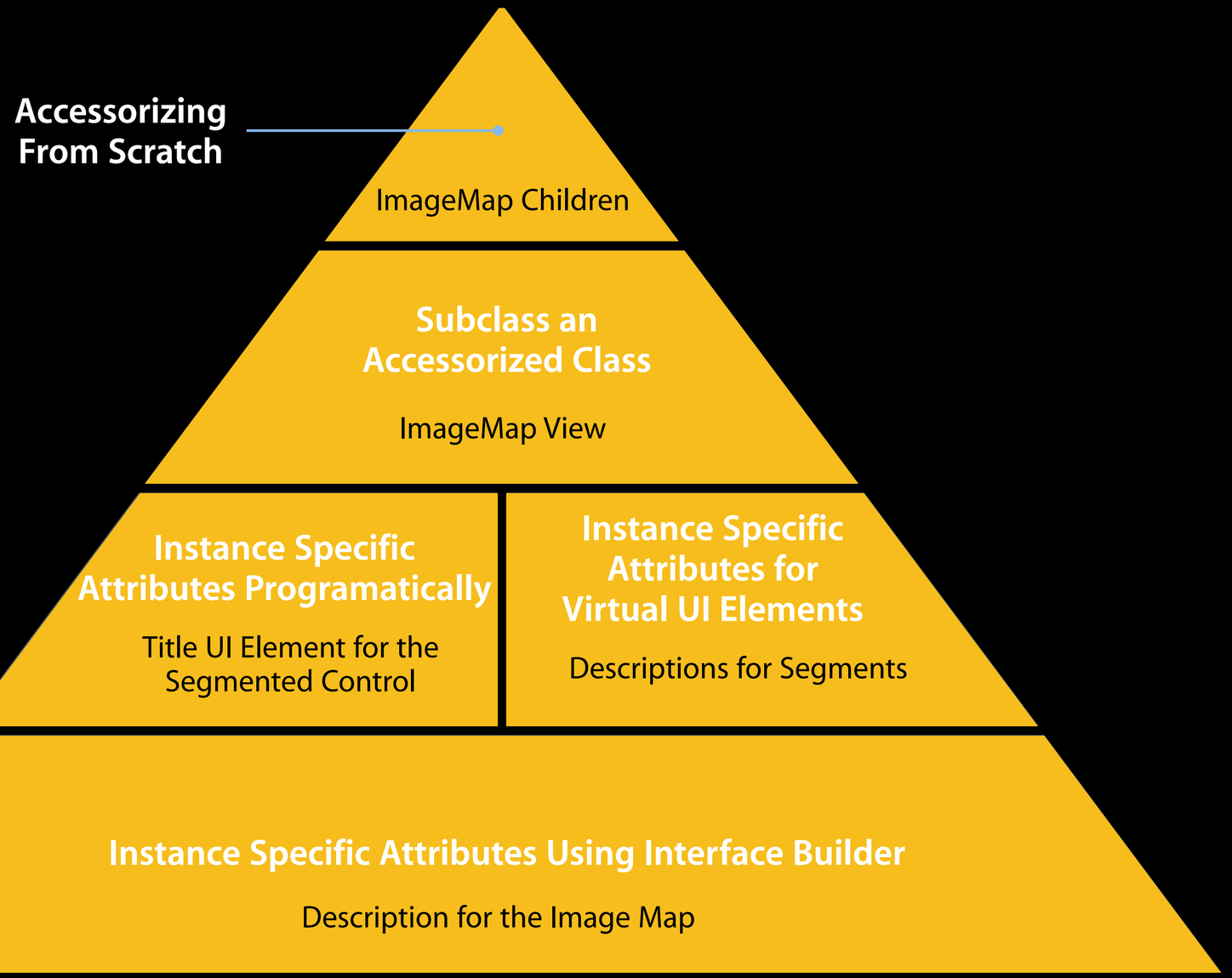
Demo

Image Map Example

http://developer.apple.com/wwdc2005/

Effort to Support Accessibility

Accessibility Pyramid



Instance Specific Attributes Using IB

Description for the Image Map

Food Pyramid

Instance Specific Attributes Using Interface Builder

Description for the Image Map

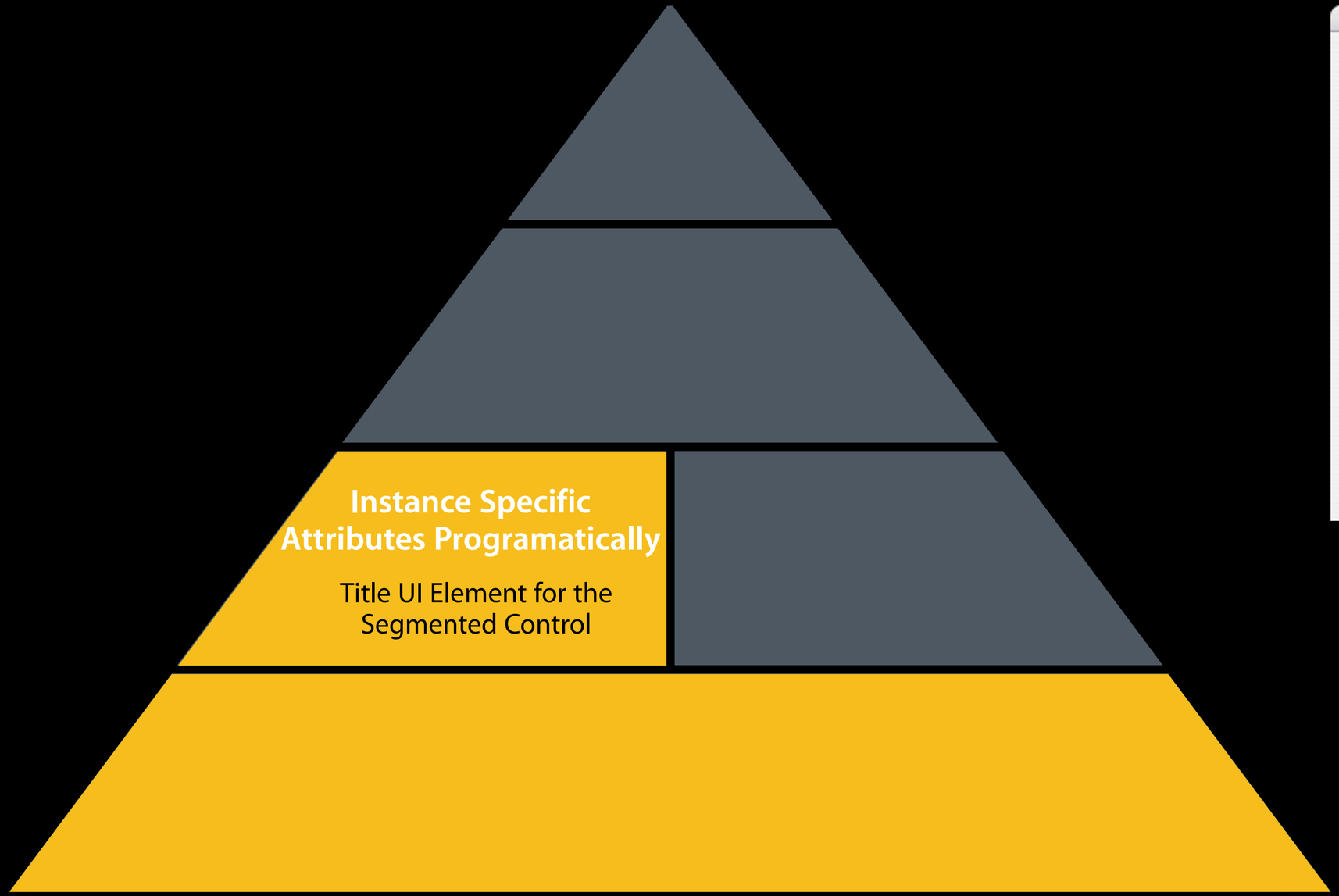
Instance Specific Attributes
Using Interface Builder

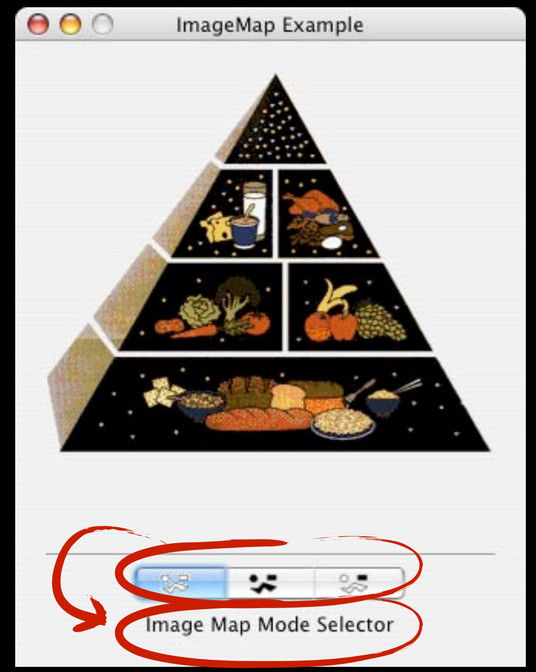
Description for the Image Map

http://developer.apple.com/wwdc2005/

Instance Specific Attributes Programatically

Title UI Element for the Segmented Control





Title UI Element for the Segmented Control

- Title UI elements actually are supported by IB
- Can still set them manually
- Use NSAccessibilityUnignoredDescendant to get the right object from the outlets
- accessibilitySetOverrideValue:forAttribute:

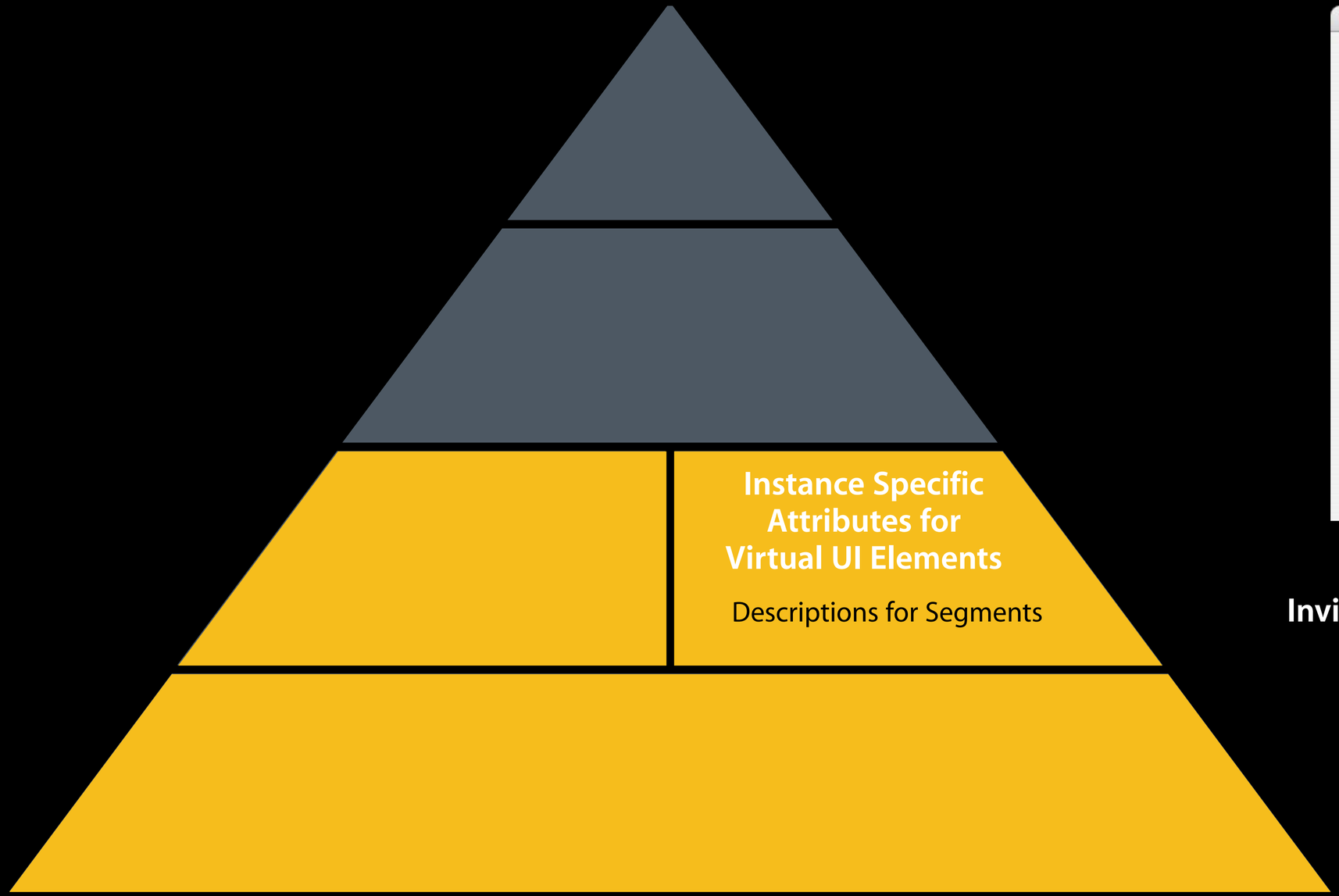
Instance Specific Attributes Programatically

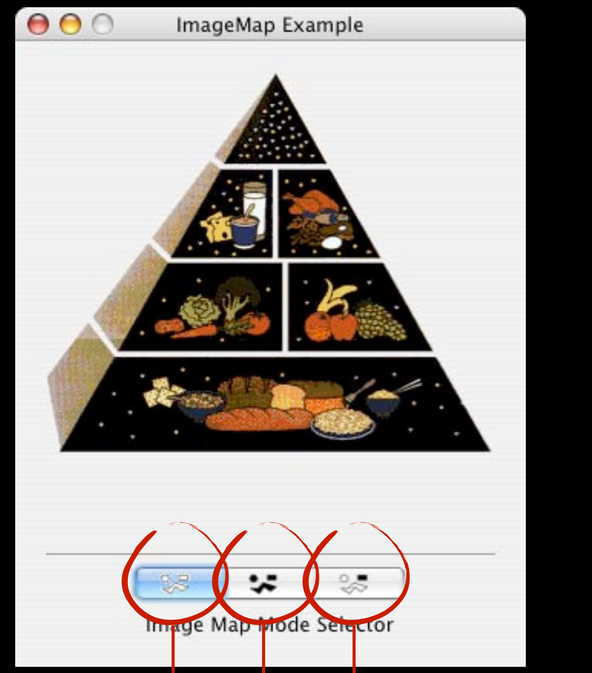
Title UI Element for the Segmented Control

http://developer.apple.com/wwdc2005/

Instance Specific Attributes for Virtual UI Elements

Descriptions for Segments





Invisible Hot Spots

Visible Hot Spots

Rollover Highlighting

AXDescriptions for Segments

- Use the NSAccessibilityProtocol to get the UI elements
- accessibilityValue:NSAccessibilityChildrenAttribute
- accessibilitySetOverrideValue:forAttribute:

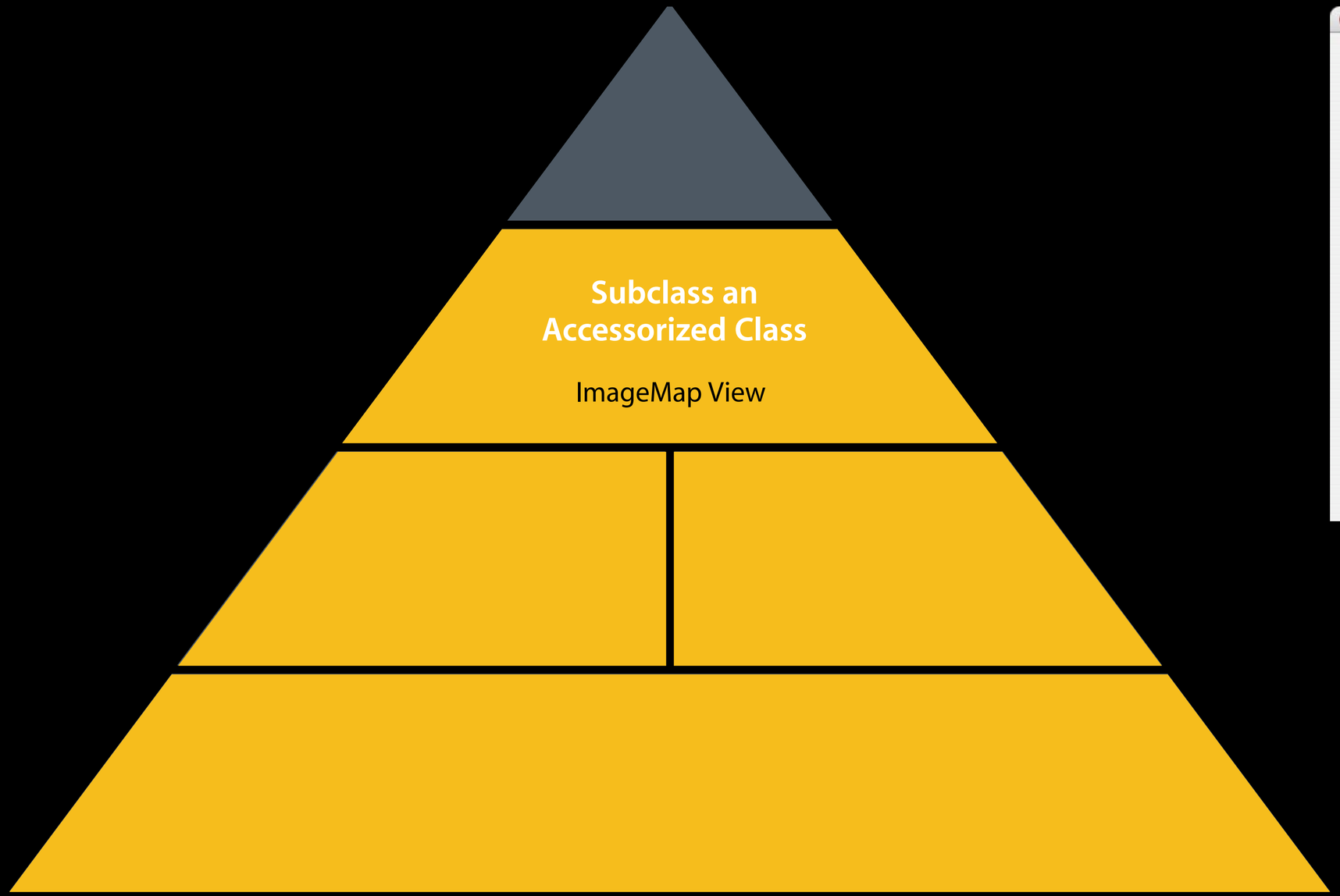
Instance Specific Attributes for Virtual UI Elements

Descriptions for the Segments

http://developer.apple.com/wwdc2005/

Subclass an Accessorized Class

ImageMap View



Accessorizing ImageMap

- Inherits from NSView
- -(BOOL)accessibilityIsIgnored
- -(id)accessibilityAttributeValue:(NSString *)attribute
- -(id)accessibilityHitTest:(NSPoint)point

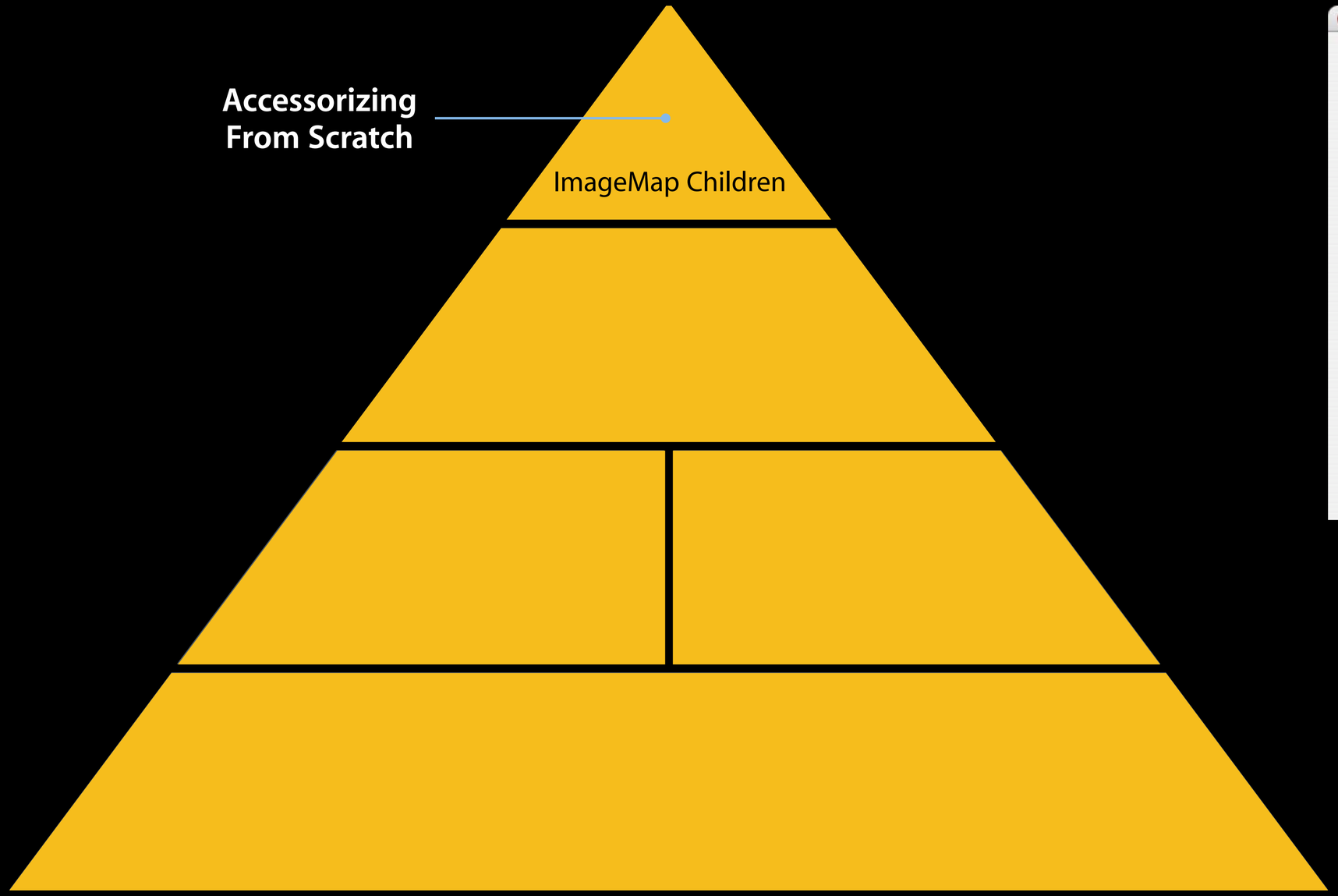
Subclassing an Accessorized Class

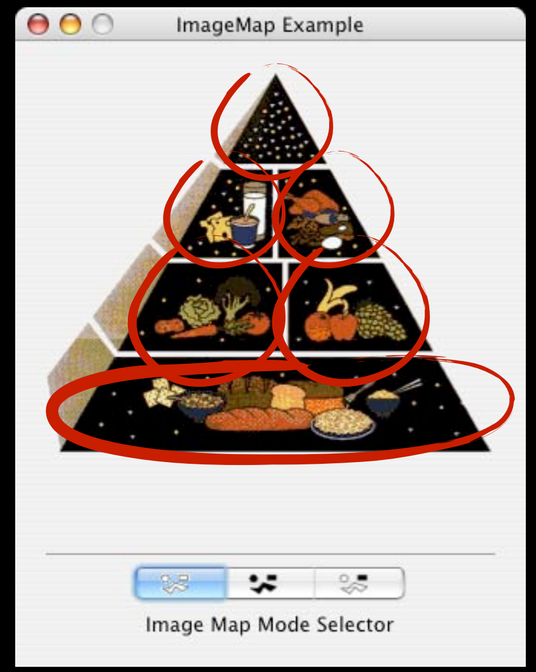
ImageMap View

http://developer.apple.com/wwdc2005/

Accessorizing From Scratch

ImageMap Children





Children of Image Map

- No existing objects to represent each hot spot
- Invent FauxUIElement
- Create as needed

What as a UIElement?

- Any NSObject that implements the NSAccessibilityProtocol

FauxUIElement

- Inherits from NSObject
- Implements NSAccessibilityProtocol
- Only keeps track of role and parent
- Relies on FauxUIElement protocol in parent to handle keyboard focus and bounds
- isEqual compares role and parent

Accessorizing FauxUIElement

- Implement ten methods of NSAccessibilityProtocol
- Four attribute methods
- Three action methods
- -(BOOL)accessibilityIsIgnored
- -(id)accessibilityHitTest:(NSPoint)point
- -(id)accessibilityFocusTest

Accessorizing From Scratch

FauxUIElements

http://developer.apple.com/wwdc2005/

HotSpotUIElement

- Inherits fromFauxUIElement
- iVar to identify which hot spot it is
- Overrides isEqual
- Adds a description attribute
- Adds a press action

Accessorizing From Scratch

Children of ImageMap View
HotSpotUIElements

http://developer.apple.com/wwdc2005/

Troubleshooting Accessibility

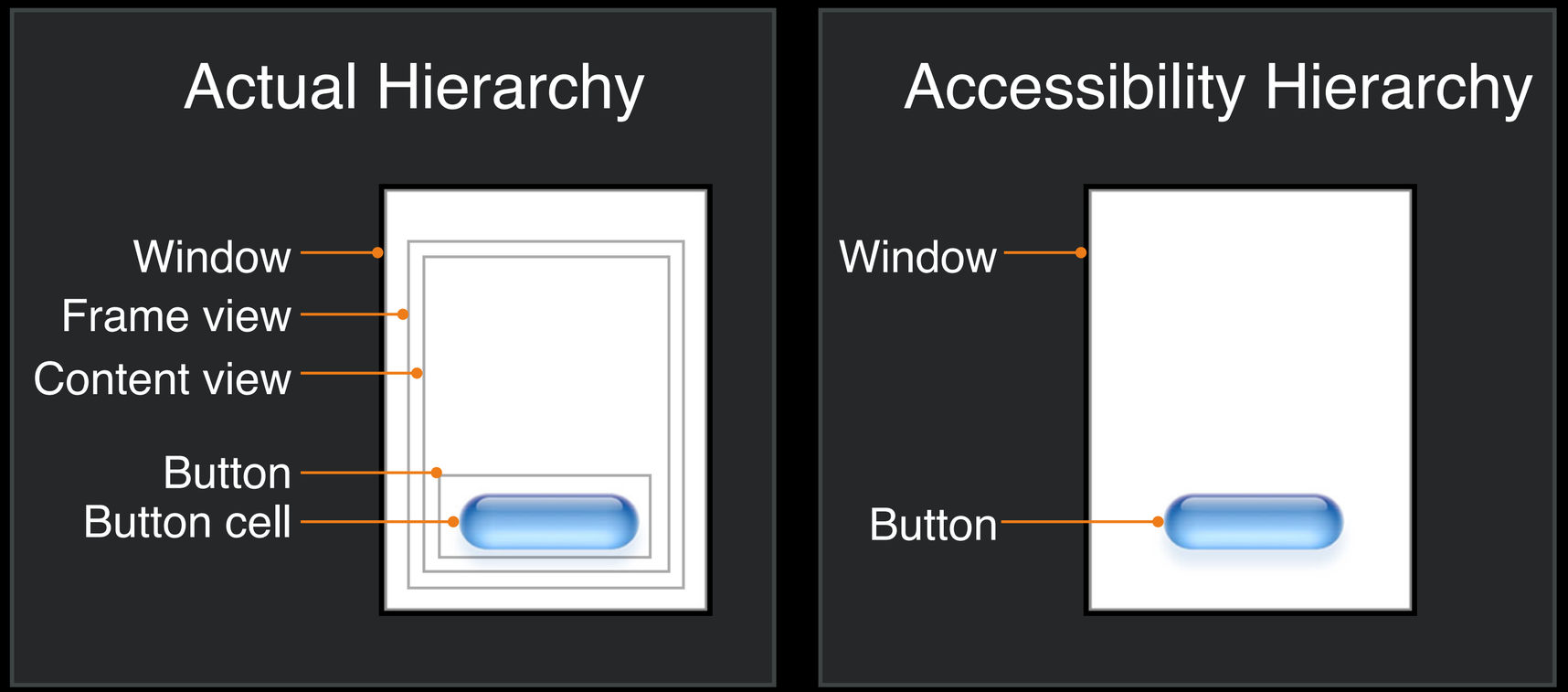
Common Errors

http://developer.apple.com/wwdc2005/

NSAccessibilityDebugLogLevel

- Launch your app with -NSAccessibilityDebugLogLevel 1
- Look for error messages in the console

Ignored UI Elements



Forgetting to Override accessibilityIsIgnored

- Symptom: “it isn’t there”
- NSView defaults to being ignored

Using an Ignored Object

- Symptom: “it didn’t work”
- E.g. single-celled controls are usually ignored—the meat is in the underlying cell
- Use NSAccessibilityUnignoredDescendant
- Use NSAccessibilityUnignoredAncestor
- Use NSAccessibilityUnignoredChildren
- Use NSAccessibilityUnignoredChildrenForOnlyChild

Failure to Implement Hit Testing

- Symptom: “child isn’t there”
- It’s easy to forget to implement hit testing

Parent-Child Asymmetry

- Symptom: “child isn’t there”
- If a child reports a UI element as its parent then that parent must report that child in its children list

Not Overriding isEqual on Subclasses of FauxUIElements

- Symptom: “only the first child works”
- Children with the same role and parent are indistinguishable

Not Taking Into Account Is Flipped

- Symptom: “it’s upside down”
- Accessibility always uses bottom-left screen coordinates
- A UIElement’s position is the bottom-left corner
- A flipped view’s origin is the top-left corner

More Information

http://developer.apple.com/wwdc2005/

Documentation, Sample Code, and Other Resources

http://developer.apple.com/wwdc2005/

Related Sessions

Accessibility and VoiceOverFeedback Forum	Session 127 Thursday 9:00AM
Accessibility Lab	Friday 2:00PM

Who to Contact

Mary Beth Janes
Sr. Segment Manager
mjanes@apple.com

Accessibility-Dev Mailing List
accessibility-dev@lists.apple.com

Q&A

Mary Beth Janes
Sr. Segment Manager
mjanes@apple.com

http://developer.apple.com/wwdc2005/

WWDC2005

Worldwide Developers Conference